Report on the 2010 Semantic Technology Conference

June 21-25, San Francisco, CA

Jody L. DeRidder

The Semantic Technology Conference is one of the few regular gatherings in the United States where institutions and organizations interested in leveraging semantic tools can obtain updates and information about the newest tools and trends in the field. This was the sixth meeting of this conference, and attendance was up markedly from last year, indicating a resurgence of interest in improving the capabilities of search, retrieval, access, and reuse of digital content in the wake of the burgeoning quantities of online information. One of the keynote speakers noted that a recent survey indicated that information workers spend 53% of their time trying to locate information[1]. With the application of semantic technology, we will be able to reclaim much of that costly staff time for use in solving problems and providing service instead.

This intense conference ran from Monday afternoon to late Friday, with tutorials at beginning and end, face-to-face meetings sandwiched in between sessions that ran from 7:30 am to often after 8 in the evening, broken only by quarter-hour breaks and often running through the lunch period. Several tracks were offered, ranging from the heavily technical to the broad theoretical overview, and conflicting time slots prevented my attendance at all of the presentations of interest. As several, including a day-long tutorial, focused on leveraging Drupal capabilities to incorporate semantic technology to best effect (and our website is now Drupal-based), I shared the presentation material for these with our Head of Web Services, in the hopes that they will prove useful and timely. At the same time, my own focus was on learning the skills, techniques, tools, and options available for optimizing access and use of our digital library holdings, so that will be the focus of this report. Again, as I was unable to attend all the presentations of interest, and as my flight was delayed, precluding my ability to attend anything on Monday, some of what I share here is drawn from the presentation material and from online documents and software referred to during the conference. Additionally, I will only share information that I believe is potentially most valuable to support UA Libraries in meeting our stated goals, and thus will not recap every presentation attended.

"Semantics for the Rest of Us"

Eric Axel Franzon, Vice President of Semantic Universe / Wilshire Conferences

Whereas Web 1.0 was about linking documents and Web 2.0 was about linking people, Web 3.0 is about linking data. This will be accomplished by uniquely identifying things and relationships and the triples that tie them together. If one thing has a relationship to another thing (for example, "a book has a title", tying together the identification of the things ("book" and "title") and the relationship ("has a") into a standardized "triple" terminology enables computers to reason with the encoded information to answer questions, infer relationships, and provide better access. The Linked Open Data movement focuses on making semantically-encoded data freely available via the web so it can be incorporated into our applications now, leveraging more information to make our own content far more usable.

1 David Siegel, "Pull: The Business Shift Behind the Semantic Web," keynote presentation at the 2010 Semantic Technology Conference, 24 June 2010.

"What Will We Be Saying About Semantics This Year?"

Dave McComb, Semantic Arts

Semantic technology is about using software to leverage our understanding and use of information. Historically we've been very dependent upon relational databases. To get the data out, you need to know the table and column in each database, which doesn't scale, as everyone has their databases organized differently. If, however, we have a single standardized way to assert facts ("triples"), we can link up data, creating graphs based on the unique identification of each thing, which allow computers to extrapolate out to put together information in new ways, uncovering new meanings and relationships. While currently we focus on humans assigning things to categories, where each thing gets a single primary category, and stays there, in the Web 3.0 approach, any thing may have multiple categories simultaneously, which makes it far more useful and accessible. Besides, there simply aren't enough humans to analyze and categorize everything; nor can we all agree on a single categorization. By developing a linked "Web of Things" we give computers the ability to provide meaningful, functional access and retrieval of content over tremendous quantities of data that humans simply cannot wade through alone.

"Introduction to Semantic Web Technologies"

Ivan Herman of the World Wide Web Consortium(W3C).

Information, to be widely useable, needs to be available on the web in a standardized structured format ordered in such a way that computers can use it: it must be mapped onto an abstract data representation which capture the relations between the bits of data. The basis for this abstract data representation is RDF (Resource Description Framework).[2] The RDF triples label the connection between the resources: (s, p, o) where s = source, p = property or relationship, o = object, using URIs [3] (including XPATH[4] queries against an XML[5] file) for each portion, or potentially literals for some of the nodes (not the relationship or property). It can be serialized in XML, Turtle[6], n3[7], and RDFA[8]. RDF triples form a directed, labeled graph. By connecting matching nodes of information from various sources, information can be merged that come from disparate databases of content.

Adding information to the merged data improves what you can do to it: if upon examination, two other nodes mean the same, or two different relations point to the same information, you can do a much better merge of the information, improving the sophistication of the queries which can be successfully made upon the data. This is what users do when they surf the web – take a bit of information here, a bit of information there, and weave it together into an answer to their questions. However we hare adding rigor to the process. By adding knowledge to the merged data sets, such as geographical information, library classifications, ontologies and extra rules, you add far more value. In practice, a model object is created; the RDF file is parsed and results are stored in the model, and then the model offers methods for retrieval. Another fairly simple application is to reuse older data; for example, to export key facts from XML as RDF and use a faceted browser to visualize and interact with the result.

---

2 W3C Semantic Web, "RDF: Resource Description Framework," http://www.w3.org/RDF/ .
3 Network Working Group, "Uniform Resource Identifiers (URI): General Syntax," http://www.ietf.org/rfc/rfc2396.txt .
4 W3C, "XML Path Language (Xpath)," http://www.w3.org/TR/xpath/ .
5 W3C, "Extensible Markup Language (XML)," http://www.w3.org/XML/ .
6 W3C, "Turtle – Terse RDF Triple Language," http://www.w3.org/TeamSubmission/turtle/ .
7 Tim Berners-Lee, "Notation 3," http://www.w3.org/DesignIssues/Notation3.html .
8 W3C, "RDFa in XHTML: Syntax and Processing," http://www.w3.org/TR/rdfa-syntax/ .

RDF Schemas (RDFS)[9] add another level, defining the terms you can use, the resources and their classes, and supporting subclasses. Thus if there is a relationship in the RDFS which applies to the RDF in the original data, you can infer that and the RDFS environment can return that triple too, thus increasing the usability of your data.

How do you get and create RDF data? Writing it manually just doesn't scale. Two solutions have emerged: using microformats[10] to convert the content into RDF, or adding RDF-like statements directly into XHTML[11] via RDFa. Most information on the web today is stored in databases. While there are bridges out there to relational databases, they are not standardized; W3C is working on a standard in this area, which may be out in April. There are also no standard vocabularies for provenance yet – W3c will issue a report in September on existing situation and then will determine whether a standardized vocabulary needs to be developed. For XML/XHTML stores, GRDDL[12] and RDFS can be used to extract RDF.

The Linked Open Data (LOA)[13] effort has the goal of exposing open data sets in RDF, setting RDF links among the data from different data sets and then if possible, query the endpoints. Their starting point is Dbpedia[14], which is a dump of Wikipedia[15] into RDF, done every 3 months. On every Wikipedia box, there's a structured box of information on the right; Dbpedia takes that, and some part of the description for an abstract, and creates RDF data from it. Geonames[16] is another database from which content is extracted and is linked in with relations such as sameAs.

To run complicated queries over RDF, SPARQL[17] was developed, with the fundamental idea of using the graph patterns and filtering the results. Multiple data sources can be specified via URI's, effectively merging the results on the fly. SPARQL is used as a service on the web, as large data sets often offer SPARQL endpoints, but we still need a standard for how to send the queries across the web. Version 1.1 is not yet finalized; it will allow updates to the data itself, not simply query functionality.

Data integration needs agreements on terms, categories used, and the relationships between them. Languages provide those things. RDFS is enough for many vocabularies but not for all; more context may be needed. It's important to balance complexity against the requirements of the application; complex tools come with a price tag. In practice, 3 technologies have emerged in the past few years: SKOS (Simple Knowledge Organization System) [18], to reuse thesauri, glossaries, etc.; OWL(Web Ontology Language) [19], to define more complex vocabularies with a strong logical underpinning; and RIF (Rule Interchange Format)[20], a generic framework to define rules on terms and data.

SKOS provides a basic structure to create an RDF representation of the data, a bridge between the print world and the semantic web. SKOS can be used to organize tags, annotate other vocabularies, etc. The Library of Congress (LOC) has assigned a URI for every term in its classifications, and they added structure (broader term, narrower term, etc.) SKOS concentrates on the concepts only; there is no characterization of properties in general. Few inferences are possible.

More complex implementations may require disjoint or equivalence of classes, and to be able to

---

9 W3C, "RDF Vocabulary Description Language 1.0: RDF Schema," http://www.w3.org/TR/rdf-schema/ .
10 "About Microformats," http://microformats.org/about .
11 W3C, "XHTML 1.0 The Extensible HyperText Markup Language," http://www.w3.org/TR/xhtml1/ .
12 W3C, "Gleaning Resource Descriptions from Dialects of Languages (GRDDL)," http://www.w3.org/TR/grddl/ .
13 "Linked Data – Connect Distributed Data across the Web," http://linkeddata.org/ .
14 "Dbpedia," http://dbpedia.org/About .
15 "Wikipedia," http://www.wikipedia.org/ .
16 "Geonames," http://www.geonames.org/ .
17 W3C Semantic Web, "SPARQL Query Language for RDF," http://www.w3.org/TR/rdf-sparql-query/ .
18 W3C Semantic Web Activity, "SKOS Simple Knowledge Organization System," http://www.w3.org/2004/02/skos/ .
19 W3C Semantic Web, "Web Ontology Language (OWL)," http://www.w3.org/2004/OWL/ .
20 W3C, "RIF Overview," http://www.w3.org/TR/rif-overview/ .

construct classes, not just name them, providing characterization of properties, more complex classification, ability to reason about some terms, etc. For this level of work, OWL (Web Ontology Language) is more appropriate. It provides an extra layer, dependent upon RDF Schemas. Whereas RDFS allows the subclassing of existing classes, in OWL you can construct new classes, enumerate valid values, and define classes through intersection, union, and complement. Various databases can be linked using these owl properties. However, very large vocabularies require great complexity. OWL is difficult; combinations of class constructions with various restrictions is extremely powerful. Full inference is difficult, and not implementable with a simple rule engine. If one abides to restrictions of a particular profile or "species" of OWL, simpler inference engines can be used. there is always a compromise between expressiveness and implementability.

In many cases, applications only need 2-3 rules to complete integration; this is what RIF (Rule Interchange Format) is about. It defines several dialects of language, and simple rule languages for the web; it is not bound to RDF only, and relationships may involve more than 2 entities. RIF allows the ability to deduce new relationships and make additional inferences. built-in data types and predicates, safeness measures, support for "forward chaining," a standard XML syntax, and a draft for expressing Core (the simplest RIF dialect) in RDF. Typically, data is available in RDF, rules on that data is described using RIF, then the two sets are "bound" and an RIF processor produces new relationships. This requires solving some technical issues, such as that RDF triples have to be representable in RIF (various constructions should be aligned, and semantics of the two worlds should be compatible).

The expressivity of OWL and RIF is fairly identical, with different emphases. The use of rules versus ontologies may largely depend on available tools, taste, local experience and expertise. There is some division in the community over one against the other. As a matter of thumb, rules are more effective for really large data sets. A bridge between the two is OWL Rule Language (OWL RL)[21]; basically it's the intersection of RIF core and OWL. Inferences in RL can be expressed with RIF rules, and RIF core engines can interact with RL engines.

"Ontology 101: An Introduction to Knowledge Representation and Ontology Development"
Elisa Kendall of Sandpiper Software & Dr. Debora L. McGuinness of Rensselaer Polytechnic Institute

Ms. Kendall began the presentation with an introduction to knowledge representation, which dates back to Aristotle. Every knowledge representation language has features such as vocabulary, syntax, semantics, and rules of inference. There are 6 dimensions by which logics can vary from classical First Order Logic (FOL): syntax, subsets, proof theory, model theory, ontology, and meta-language. Ms. Kendall went on to discuss intensional and modal logic, natural language processing, computational logic, and description logic.

A knowledge base is where you store your instance data. An ontology contains metadata; "an Ontology is a specification of a conceptualization" (attributed to Tom Gruber). Ontologies provide a shared vocabulary for use between various services. To develop an ontology, it is best to begin with canonical definitions; IDEF5[22] might be a good place to start; it provides a method of capturing an ontology from content itself.

Ms. Kendall went on to cover some basic guiding principles about using the Web Ontology Language, OWL. Most collected database metadata operates under under closed-world assumptions ("if not in my database, it doesn't exist"). Uncertainty is magnified in open-world conditions, making

21 W3C, "OWL 2 RL in RIF," http://www.w3.org/TR/rif-owl-rl/ .
22 Knowledge Based Systems, Inc. "IDEF Integrated DEFinition Methods," http://www.idef.com/Home.htm .

reasoning far more difficult. OWL is designed to overcome this problem. Ms. Kendall traced the history of OWL, then looked at description development (defining domain terms and inter-relationships); then talked about classes & hierarchies & inheritance. There are different modes of development: top down or bottom up, or some combination. (This part of the presentation became quite technical, so I will avoid further description of it here.)

The portion of this tutorial that I found most helpful was that presented by Dr. McGuinness, the creator of the infamous Wine Ontology[23] some 25 years ago (she worked at Stanford then; has also done research at Bell Labs, and is now at RPI[24]. She provided some very helpful rules of thumb for those who decide they want to write an ontology. The first rule of thumb is to reuse someone else's if at all possible, and modify it to meet your needs. Second is to avoid cycles (endless loops) by creating equivalencies. If when designing your model, you create a class which only contains a single subclass, you have to ask why the superclass was created; it may be unnecessary. You also don't want too many subclasses; instead, add intermediate layers. Think about consistency in how you relate things to each other. Stick with singular descriptive terms rather than plurals. Synonym names are part of the class definition, not different classes. When determining class versus property values, listen to your target audience audience and subject experts, because it's a judgment call. Then be consistent.
Where do you stop with your class hierarchy? If you hold it in your hand, it's an instance. Otherwise, it's a class.

Limit the scope. Go by the use case: what questions do you want to answer? What tools have you used, and why didn't they work? There is no need for objects to have 40,000 properties unless there's a really good reason. Getting consensus from stakeholders is difficult – use use cases. What is the question that they're trying to answer, and where did that make a difference? You need to make sure non-experts can maintain the ontology. Avoid unnecessary complexity and extraneous information (think mobile phones).

Also, don't hand-build the first ontology. Protégé[25] is a good tool, and there are others as well. Must be syntactically correct. Do consistency checking; there are some tools out there for this as well. Good commercial off-the-shelf tools include Pellet[26], RacerPro[27], FaCT++[28], KAON2[29], and VIStology's ConsVISor OWL consistency checker[30].

Dr. McGuiness recommends that one not go beyond OWL DL or Lite[31], in order to support inference and reasoning. Run the model through consistency checkers and reasoners, to look for problems. Check the instance information and do some testing, then embed and distribute it.

Don't do massive ontology development without knowing your starting point. Find your best starting point: what ontologies are out there, and what are your users already using? It may be best to clean up, integrate and extend what they are using, though it is far from the best available.

The semantic web community has reached a consensus that there will be no single upper-level ontology. Instead, there will be multiple ontologies that you have to integrate. Use the tools, then use

---

23 Knowledge Systems, AI Laboratory, Stanford University, "How does it work? (OWL Example Wine Agent)," http://www-ksl.stanford.edu/projects/wine/explanation.html .
24 Rensselaer Polytechnic Institute, http://rpi.edu/ .
25 Stanford University, Protégé (Ontology Editor), http://protege.stanford.edu/ .
26 Clark & Parsia, "Pellet: OWL 2 Reasoner for Java," http://clarkparsia.com/pellet/ .
27 Franz, Inc, "RacerPro," http://www.franz.com/agraph/racer/ .
28 Dmitry Tsarkov and Ian Horrocks, "FaCT++," http://owl.man.ac.uk/factplusplus/ .
29 Boris Motik, "KAON2," http://semanticweb.org/wiki/KAON2 .
30 VIStology, Inc., "ConsVISor: A Tool for Checking Consistency of OWL Ontologies," http://173.14.188.57:8080/consvisor/ .
31 W3C, "OWL Web Ontology Language Overview," http://www.w3.org/TR/owl-features/ .

your brain to analyze the results of the tools.

Sometimes the best ontologies are very simple, and these are used by many people, such as FOAF (Friend Of A Friend)[32]. How important is the risk of misuse for inferences as opposed to the cost of complexity?

A big issue is how to know whether to trust the information found as valid. An explanation system (inference web) can provide provenance information such as information on the knowledge source. Trust and understanding are built by transparency, providing information about how you do something, why you make what decisions, and how you manipulate data. Proof markup language (PML)[33] is a new kind of linked data on the web. It's focus in on provenance, justification, and trust: it annotates provenance properties, encodes provenance relations, and adds trust annotation.

When combining two or more sets of data, one must set up matching between the different ontologies in order to merge them; how can users trust that the matching was appropriate? While working for Stanford University, Dr. McGuinness helped developed IWTrust[34], a method of providing clear explanations and proofs of the mappings created. Trust can be inferred from a web of trust. This infrastructure includes a trust component responsible for computing trust values for answers.

Dr. McGuinness described an NSF project for the Virtual Solar Terrestrial Observatory[35], in which a small team pulled off a very useful, extensible ontology in 8 months, which is still in use. She also described the Scientific Observations Network (SONet)[36], a community-driven effort to achieve semantic interoperability of environmental and ecological data. They needed to build a community-sanctioned, evolving data model for observational data., a network for practitioners, and a repository of motivating use cases, in order to enable interoperability of existing data.

A fascinating new development underway, funded by DARPA[37], is CALO: Cognitive Assistant that Learns and Organizes:"The goal of the project is to create cognitive software systems, that is, systems that can reason, learn from experience, be told what to do, explain what they are doing, reflect on their experience, and respond robustly to surprise."[38]

"Using a Controlled Vocabulary for Managing a Digital Library Platform"

Sean Boison of Logos Bible Software

Mr Boison has merged content from 7 Bible dictionaries, done a conservative automatic alignment, and reviewed it manually to reduce over 40k subject-oriented concepts down to about 10,000. He is currently adding additional resources, which suggest new concepts and alternate terms. The use case is information discovery: the software now automatically links reference to concepts, concepts to related concepts, and concepts to references. Mr. Boison is also text mining for reference to concepts. Each article votes on most likely references and most likely concepts for a reference. He's creating a reverse index from reference to concepts. Estimates should improve with more content. He also extracts and aggregate key terms, weights for relevance and provides clustering of documents. In the future he plans

32 Dan Brickley and Libby Miller, "FOAF Vocabulary Specification," http://xmlns.com/foaf/spec/ .
33 Deborah L. McGuinness et. al, "Proof Markup Language," http://inference-web.org/2007/primer/ .
34 Ilya Zaihrayeu, Paulo Pinheiro da Silva, Deborah L. McGuinness, "IWTrust: Improving User Trust in Answers from the Web," http://www.ksl.stanford.edu/people/pp/papers/Zaihrayeu_iTrust_2005.pdf .
35 National Center for Atmospheric Research, "Virtual Solar Terrestrial Observatory," http://www.vsto.org/ .
36 "SONET: Scientific Observations Network," https://sonet.ecoinformatics.org/front-page .
37 DARPA: Defense Advanced Research Projects Agency," http://www.darpa.mil/ .
38 SRI International, "CALO: Cognitive Assistant that Learns and Organizes," http://caloproject.sri.com/ .

to add linking into LCSH and WordNet (a freely available online thesaurus)[39] .

"Semantic Web for the Working Enterprise"

Dean Allemang of Top Quadrant and James Hendler of Rensselaer Polytechnic Institute.

These presenters showed a number of short video clips about the semantic web and discussed and rated them as if they were movie reviewers, which was quite entertaining. Per "Tim Berners-Lee on the Next Web"[40], with his call for "Raw Data Now", they rated his presentation as good on vision and weak on short term ROI (Return on Investment). Highly rated was Neal Goldman's (CEO of Inform) "Making Sense of the Semantic Web"[41], which urged us to pay attention to RDFa, already in use by Facebook, BestBuys, and Google's Rich Snippets. Linked data relies on good URIs. RDFa is very simple and very powerful. It has emerged as the way to do it (to get your content out there in a semantically available way). It was designed for the web, for how enterprises are already organizing their content.

The next video was by Clay Shirky and David Weinberger on "Web 3.0".[42] Dean disagrees with Shirky about one thing: the semantic web is *not* Artificial Intelligence. It should *not* occur to people that there is only one way to organize stuff. You don't need to have a single way to organize the web. We are moving from a common view to many consensus views that link to one another: ontology libraries that interlink. If you publish things in RDF, they need to have deep meaning, via reuse. Reuse is what it's all about. Shirky formed a wrong view of the semantic web 8 years ago, arguing for an alternative that isn't realistic. Still, both presenters gave a thumbs-up to this video. We need good semantics, but do not need heavy-weight ontologies. Underpinnings need to be right, but we wasted lots of time on model theory. The user of the stuff need not know about all of that. (For more on this topic, see the book these two presenters wrote. [43]).

"**True Semantic Reasoning: Self-Constructing Ontologies"**

Oliver "Olly" Downs of Atiego, Jeff Jonas of IBM and Marc Davis of Invention Arts

This very practical presentation focused on leveraging the capabilities of our cell phones to automatically capture time and location of everything we do online. This automated meta-tagging serves to disambiguate online information, and can be used to make inferences, as two people cannot be in the same place at the same time, nor can one person be in more than one place at a time. By adding time and location to data, we tag our online lives and ground online data in reality, disambiguating information and building timelines and trajectories which can be both captured and studied.

We must have fundamentals, without ontologies. There is structure in the world: time and space, entities and rules. Build the assumptions into the system. Below that, you don't need much. Emergent semantics come from situating the data in the graphs. For things to scale, you have to have some

---

39 Princeton University, "WordNet: A lexical database for English," http://wordnet.princeton.edu/ .

40 TED Conferences, LLC, "Tim Berners-Lee on the Next Web," http://www.ted.com/talks/tim_berners_lee_on_the_next_web.html .

41 Bloomberg Businessweek, "The Semantic Web: Making Sense of the Semantic Web", Neal Goldman, http://feedroom.businessweek.com/index.jsp?fr_story=5c4c05ae279f6d20a17636fdb706daea352b1955 .

42 Kate Ray, "Web 3.0 a doc by kate ray", Clay Shirky and David Weinberger, http://kateray.net/film/

43 Dean Hendler and James Allemang, Semantic Web for the Working Ontologist, (Elsevier Science Ltd, 2008) http://www.amazon.com/Semantic-Working-Ontologist-Hendler-Allemang/dp/B001E3O2K6

certainty. Data is one thing, user queries is something else, reference data is something else – what if it is all considered data? Put it together and cool stuff happens. There are rights of ownership of data – you trade it with the internet. We need to change how we interact with data about ourselves. The digital realm is out of sync with the real world in terms of rights; but rights and ownership of personal data is going to change incredibly soon.

It's depth that matters; we need density and divergence (different data that you can converge). Then you can collapse it into meaningful associations. If data is sparse and density is low, it takes much longer, not much to work with.

"The Ten Myths of the Semantic Web Debunked (and how they hurt your success)"

John Hebeler of BBN Technologies, and Matthew Fisher of Progeny Services

I found this both fascinating and very helpful. The ten myths they identified are as follows:

1) "Traditional data management is good enough." This myth forces expensive approaches to jury rig old technologies to fit the new realities, and you lose the opportunity to leverage your information. The older methods do not provide support for integration, conceptualization, disambiguation, or relevance, nor do they support scale and adaptation.
2) "Cool Web 2.0 is good enough & boring SOA [Service Oriented Architecture] is good enough." This belief provides little value, and is a poor use of data and services, though it provides great semantic potential. Though thousands of APIs (Application Programming Interfaces) exist, only 10 hold 95% of all mashups; most listed are not used in even a single mashup. Most mashups only combine 2 potential APIs, and few go beyond 5.
3) "Publishing your data is good enough." A tremendous amount of effort is spent in getting data out there at the expense of making it truly useful. It makes it very difficult to find useful information and access it. Even the government is moving to RDF.
4) "Semantic web doesn't work." Those who believe this are delaying adoption and losing a critical advantage. The semantic web has matured, and it's getting easier to use. sameAs.org[44] has over 10 million bundles, dbpedia.org has over 1 billion triples. Newsweek, NY Times, Yahoo, Facebook and more have developed semantic web capabilities. It works. It's growing.
5) "Ontologies are the answer." Many semantic web solutions do not even have an ontology. A fully fleshed out ontology takes infinite time and cost; simple ontologies often serve better. Data is key, and more data beats more algorithms. Ontolgogies by themselves don't do anything.
6) "Inference is the answer." Inference often performs poorly at large scale; it works best when it can focus on a subset of data. Many semantic web solutions use little or no inference. It is cool, but it's not the answer.
7) "Semantic Web forces you to recreate all your data." Believing this myth causes a delay in adoption and loss of critical advantage. Simply adding a semantic layer over current content brings semantics to non-semantic data.
8) "Semantic Web is Natural Language Processing (NLP)." Believing this myth sets up the wrong expectations and tries to use semantic web technology to solve the wrong problems. Semantic

44 "<sameAs>: interlinking the Web of Data," http://sameas.org/ .

web consists of three main items: Knowledge Representation including some logic, tools, and knowledge bases.

9) "Semantic Web handles all your information needs." This myth raises expectations too high; no technology ever solves everything. Semantic web adds value to existing sources, working with other technologies including databases, XML web services, and proprietary technologies and data sources.
10) "Ontologies merely create a new information island/stovepipe." Hand-carved ontologies are often the quick short term solution with a long term cost. There are thousands of useful ontologies available today covering both general and specific domains. If ontologies are needed, existing ones need to be reused as much as possible.

The semantic web offers a viable solution to information overload, enabling information acceleration (and hence value), partnering with existing data and services, making "more" better. It should be incorporated iteratively and incrementally.

"Basic Level Categories for Knowledge Representation: Combining Cognitive Science and Library Science"

Tom Reamy, KAPS Group

This presentation may provide some guidance in our current exploration of providing category browse options for our digital collections. Mr. Reamy expounded on how to identify what level of category is basic, and to whom such a level of category appeals. There are three levels of categories: superordinate, basic, and subordinate. (Examples would be: mammal: dog: Golden Retriever, and furniture: chair: kitchen chair.) The general populace prefers the basic level, though entire societies and communities differ in their preferred levels. Expert users do not make use of basic categories, for they have already specialized beyond those in their fields. A philosopher will rarely, for example, use the term "philosophy" for what interests him is specialized subsets of that basic realm. Experts use specialized language, based on deep connections; novices prefer higher, superordinate levels.

Basic level categories are those which children learn first and most easily, usually short words with maximum distinctness and expressiveness. They are mid-level in a taxonomy/hierarchy, usually used in a neutral context, and this is the level at which most of our knowledge is organized. Basic categories are composed of the most commonly used labels – with objects, they are the most similarly perceived shapes. For non-objects (concepts) it's much more difficult, as there is no widespread consensus, nor is there a clear hierarchical relationship between concepts.

One way to recognize basic level categories in a corpus of content is to pull out short words and noun phrases, then pull out the stop words. Another way is to look at the attributes associated with terms. Terms are usually superordinate if they have functional attributes, basic if their attributes are nouns and adjectives, and subordinate terms are often associated with adjectives. Basic level is also often context-dependent. One tool you can use is Cue Validity – the probability that an object belongs to some category given that it has a particular feature (for example if it has wings, it is probably a bird unless the corpus is about airplanes or dragons). Superordinates have lower and fewer common attributes. Subordinates share more attributes with other members at the same level.

In order to analyze a large corpus of content, first divvy up some documents according to what is expert and what is basic. Then give it to the computer, have it analyze the text and come up with some

rules; then you then weed out about 80% of what it came up with. You'll need to add in operators and massage the terms it comes up with to fine-tune the terms, and build in sophisticated rules.

Application areas for this include taxonomy development/design: use the basic level. As far as user contribution, tests in card-sorting have shown that non-experts use superficial similarities; it doesn't work well. Experts come up with different similarities. Experts come up with tags that are good for other experts, not for the general public. It's better to ask users for attributes instead; or or you can develop different versions by the community (expert or general public). A possibility is to present the combination of superordinate and basic level -- or expose things at different levels and allow the user to move up and down easily. If using document maps – expose at the basic level, not the high level.

Remember that ontology development must focus on the intended audience. Basic categories can be used for search, relevance ranking, information presentation (tag clouds), and clustering, to enhance browsing capabilities.

"Pull: The Business Shift Behind the Semantic Web."

David Siegel, author of *The Power of Pull*

This was a brilliant and charismatic keynote speech that had a tremendous impact on the conference. Siegel predicts that in the future, we will each store all our information in our own private online location which will interact with the web to bring us exactly the information we want and need at the point we need it. It will serve as a personalized portal into the online world, and we will be in control of our own information as never before. The following is a synopsis of Siegel's presentation.

In the world of pull, you don't own the customer, the customer owns you. In the future, we will each have a Personal Data Locker -- you log in, and you have everything you want and need at your fingertips. To do it at scale will require principles of the semantic web.

Substitute "unambiguous" for "semantic." The Wolfram Alpha[45] search engine, for example, is based on unambiguous data. Semantic information is unambiguous, findable, interoperable, modular/reusable, and pullable. There are no copies and no translations.

Knowledge workers use 53% of their time, on average, finding information. When we make it more findable, we'll spend more time solving problems. Here's an example of a problem: UPS, USPS, and other shippers all do NOT interoperate, which means that every business has to follow different protocols for each one; that's a waste of time and energy.

The first thing we need to do is to assign a unique identifier to a person allows the person to move as they want. Thanks to XBRL (EXtensible Business Reporting Language),[46] information in the financial world doesn't have to ever be repeated. It uses the GAAP (Generally Accepted Accounting Principles in the United States)[47] taxonomy, and is currently in use in over 90 countries.

We want to make our information pullable. Think of data as water – in the world of pull, we are building indoor plumbing. Look at how search and retrieval is managed on the "Blue Nile" website[48]; then imagine doing the same things over all content on the web. We've gotten used to NOT comparing apples to apples, but that needs to change.

Now imagine hunting for a job in a world where all this is possible. Using a semantic query within your data locker, you would describe yourself and your desires, make them visible, and the matches

45 Wolfram Alpha LLC, "WolframAlpha: computational knowledge engine," http://www.wolframalpha.com/ .

46 XBRL International, "XBRL: Extensible Reporting Language," http://www.xbrl.org/WhatIsXBRL/ .

47 CPAClass.com, "U.S. GAAP: Generally Accepted Accounting Procedures in the United States," http://cpaclass.com/gaap/gaap-us-01a.htm .

48 Blue Nile, Inc., "Blue Nile: Education, Guidance, Diamonds and Fine Jewelry," http://www.bluenile.com/ .

come to you by machine. Once you have chosen a position, you flip the switch and you are no longer visible as looking for jobs.

On your dashboard, you manage your rights and preferences. This dashboard contains information about all you own, your money, and all you care about. You are in control of your data.

We don't need APIs. The data we're going to use already has one built in. The Pull acid test, is to determine if a project will scale up; data must be designed to be pulled, it must be compatible to other data on the web, and it must be unambiguous. Successful efforts in this regard include the New York Times, XBRL and data.gov[49].

In push, we are guessing. In the world of pull, customers have the control, and can mash up your data. You need to invert your processes, your business model. Allow account and data portability. "Move my account" will be on everyone's screen. We have about 10 more years of digitizing to go. Now we are putting our most important information into semantic notation; it will take about 30 years, and has already begun.

We can save 7 trillion dollars by moving to pull. What drives the need? Each year we go forward, there is a bigger risk in maintaining our legacy systems. We should talk about alignment – the alignment points between you, your competitors, your suppliers, and your customers.

What we do online and how we do it matters.

David Siegel moderated a panel later the same day with presentations about **"The Personal Data Locker"** and how to implement it. Clearly he has already garnered the support and created the synergy to move the "pull" model of obtaining information from a theory into the realm of practicality.

Chris Messina, a Google Open Web[50] advocate with a background in OpenID[51] and "Activity Streams" (a format for syndicating social activities around the web)[52] (and the creator of the use of the #hashtag in Twitter[53] tweets) expects easier APIs in JSON[54] to power the data locker. The semantic web technologies of yesterday get harder and harder to use – we suffer with complexity. We need low cost, easy, simple implementations; in his opinion it comes down to identity and simple APIs.

Phil Wolff managing editor of Skype Journal[55] and volunteer director of the Data Portability Project[56], states that privacy policies were created when times were simpler. Today services build on services; they depend upon transparency and disclosure. Wolff announced the imminent launch of the "Portability Policy" project[57] to encourage every site to explain their data portability practices in a data portability policy. Portability policies disclose what you enable people to do with their own data.

Marc Davis, of Invention Arts (previously at Yahoo), focused on personal data exchange. He quoted Meglena Kuneva (European Consumer Commission) as saying "Personal data is the new oil of the internet and the new currency of the digital world." So where are the banks, the exchanges, and the institutions and instruments for storing, managing, and trading this money? We are in the digital feudalism stage of the web. We are serfs, serving others; we do not yet own property or our own rights and data. Property is about having the rights to benefit from your data.

New economic and societal value will come from from "User-Centric Personal Data Aggregation."

---

49 U. S. Government, "Data.gov: Empowering People," http://www.data.gov/ .
50 "Open Web Foundation," http://openwebfoundation.org/ .
51 OpenID Foundation, "OpenID,' http://openid.net/ .
52 Diso Project, "Activity Streams: a format for syndicating social activities around the web," http://activitystrea.ms/ .
53 "Twitter: Discover what's happening right now, anywhere," http://twitter.com/ .
54 "Introducing JSON (JavaScript Object Notation)," http://www.json.org/ .
55 "Skype Journal: Independent. Inciting Innovation since 2003," http://skypejournal.com/ .
56 "DataPortability Project," http://dataportability.org/ .
57 DataPortability Project, "Your Portability Policy," http://portabilitypolicy.org/index.html .

Davis envisioned this model as follows: there will be a new personal data stack and ecosystem, built on metadata and personal data collection. Personal data banks will be supported by digital identity management and encryption. Above personal databanks will exist a layer of personal data refinement (e.g. parametrization), and above that, personal data exchanges. Above personal data exchanges is Applications, and above that is Monetization.

Drummond Reed (of the International Card Foundation[58] and Open Identity Exchange (OIX)[59]) shared five thoughts about personal data lockers, and enjoined us all to "Think about what it's going to mean for your business. It's happening now."

1. (PDS= Personal Data Store) Your personal data locker provider will be your personal identity provider. (Such as OpenID)
2. There will be a global trust framework for personal data exchange (such as OIX: Open Identity Exchange). This will largely replace the Customer Relationship Management for your company.
3. XDI is XRI (Extensible Resource Identifier[60]) Data Interchange[61]: – there will be a semantic data sharing protocol for PDX (a moniker representing an global network of Personal Data Stores[62]). This is vendor relationship management; there must be a way for the stores to talk to each other, and they will need a data sharing protocol. XDI is a close cousin of RDF, and incorporates a way to do portable permissions.
   There will be a PDX network and a network-wide PDX applications ("App") store. It's already happening. You can keep your identity cloaked until you buy.
4. There will be a new global non-profit foundation for PDX (PDX.org).

David Boardman of Atiego xPatterns[63] opines that every web experience will be more personal. Businesses would benefit by knowing as opposed to guessing what people want and need. We need to develop "me-centric" services, but to do this, we need to know: who are these people? And who gets to control this information? Right now, mainly businesses do. In the future reality, it will live somewhere between consumers and businesses. But consumers need more control. Right now these data stores are fragmented, explicit and not-actionable. They will become holistic, learned, and actionable. How? Via semantics and analytics. The software will learn from your activity, the things you own, the things you do and look at. You'll be able to correct it and edit the information it gathers. This ability to learn from you will accelerate the adoption. It will be transparent, accepted, learned, and actionable. Personal experience will drive adoption. As more technologies work on their behalf, and are transparent, consumers will will adopt them.

The remainder of the sessions I attended were either highly technical (including several hours of tutorials on implementation) or not as relevant to our needs as I had hoped, so I will forbear discussing them here.

Conclusions

58 Information Card Foundation. "Information Cards," http://informationcard.net/ .
59 "Open Identity Exchange: Building Trust in Online Identity," http://openidentityexchange.org/ .
60 OASIS, "OASIS Extensible Resource Identifier (XRI) TC," http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xri .
61 OASIS, "OASIS XRI Data Interchange (XDI) TC," http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xdi
62 Drummond Reed, "The PDX is coming," blog entry on "Equals Drummond," http://www.equalsdrummond.name/?p=297 .
63 Atigeo, "The xPatterns Story: Not a point solution but a platform," http://www.atigeo.com/xpatterns/science .

As the demand for easily obtainable information and control of one's personal information grows, the ability of our digital resources to be pulled into use by our patrons is going to be determined by the extent to which we have made it “pullable” by web agents and search engines. The more inter-linked it is, and the better we leverage semantic capabilities in a low-cost and reasonable manner, the more visible and valued our content will be. Several tools have been developed to ease the modifications of what we already do to semanticize our content, without requiring the expensive tasks of reworking our content or developing our own ontology and mapping it to those of others. With the advent of RDFa and user-friendly tools, the first level of implementation can be nearly painless.

After comparing the tools and options currently available, I recommend that we consider modifying our dynamic display of digital content to include such RDFa tags as is practical and useful. This will require some analysis on the part of the metadata librarians to ensure appropriate tagging, and modification to the templates that determine how our metadata files are rendered for the web. Additionally, I believe it would be in our best interests for our Web Services department to determine whether and to what extent the new semantic capabilities of Drupal could be leveraged to make library web content more accessible and useful.

As entity extraction and relation inference tools continue to develop, we may in the future move towards automating even deeper linking for better access to the hidden content of our textual materials. Monitoring the developments in the field will serve to our advantage, so that we can snap up the easiest, lowest-cost options as they arise, making us one of the foremost digital libraries in the world and a clear leader in our field, bringing our content to the fingertips of researchers ahead of that from any other digital repositories.